

Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Курский государственный университет»

Кафедра программного
обеспечения и администрирования
информационных систем

Направление подготовки
математическое обеспечение и
администрирование
информационных систем

Форма обучения очная

Отчет
по лабораторной работе №4
«Программирование циклических алгоритмов»
дисциплина «Основы алгоритмизации и программирования»

Выполнил:

студент группы 113.1

Козявин М. С.

Проверил:

старший преподаватель кафедры ПОиАИС

Ураева Е. Е.

Курск, 2021

Цель работы: изучить основные приемы программирования циклических алгоритмов.

Задание

Задача 1. Некий мужчина отправляется на работу, которая находится на расстоянии 1 км от дома. Дойдя до места работы, он вдруг вспоминает, что перед уходом забыл поцеловать жену, и поворачивает назад. Пройдя полпути, он меняет решение, посчитав, что правильнее вернуться на работу. Пройдя $1/3$ км по направлению к работе, он вдруг осознает, что будет не хорошо, если он так и не поцелует жену. На этот раз, прежде чем изменить мнение, он проходит $1/4$ км. Так он продолжает метаться, и после N -этапа, пройдя $1/N$ км, снова меняет решение. Определить какой общий путь пройдет мужчина.

Задача 2. Даны положительные действительные числа y_0, x, ε . В последовательности y_1, y_2, \dots , образованной по закону: $y_i = \frac{1}{2} (y_{i-1} + \frac{x}{y_{i-1}-1})$ $i = 1, 2, \dots$, найти первый член y_n , для которого выполнено неравенство $|y_n^2 - y_{n-1}^2| < \varepsilon$

Задача 3. Решить задачу 2 индивидуального варианта задания, используя цикл с постусловием.

Задача 4. Два натуральных числа называют дружественными, если каждое из них равно сумме всех делителей другого. Например, дружественными является пара чисел 220 и 284 (сумма делителей числа 220: $1+2+4+5+10+11+20+22+44+55+110=284$; сумма делителей числа 284: $1+2+4+71+142=220$). Найти все пары дружественных чисел, меньших натурального k .

Разработка алгоритма

Задача 1

Входные данные: n – действительное число

Выходные данные: sm – действительное число

Алгоритм решения задачи представлен на рисунке 1.

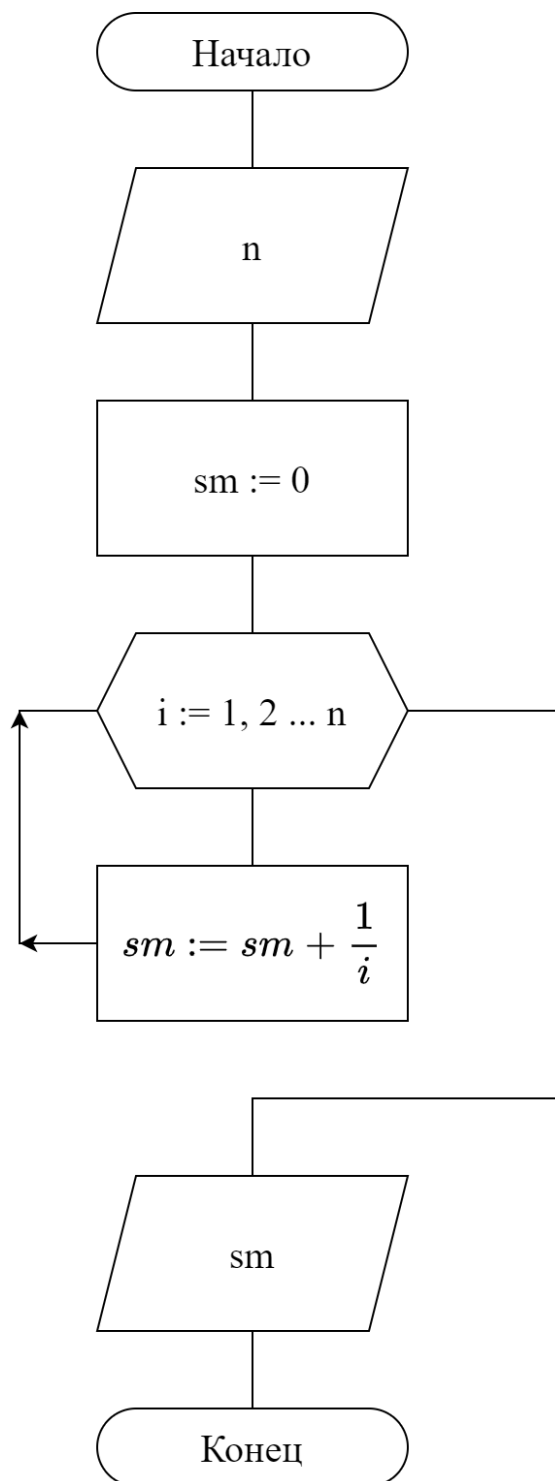


Рисунок 1 - Алгоритм решения задачи 1

Задача 2

Входные данные: $last$, x , e – действительные числа.

Выходные данные: y – действительное число.

Алгоритм решения задачи представлен на рисунке 2.

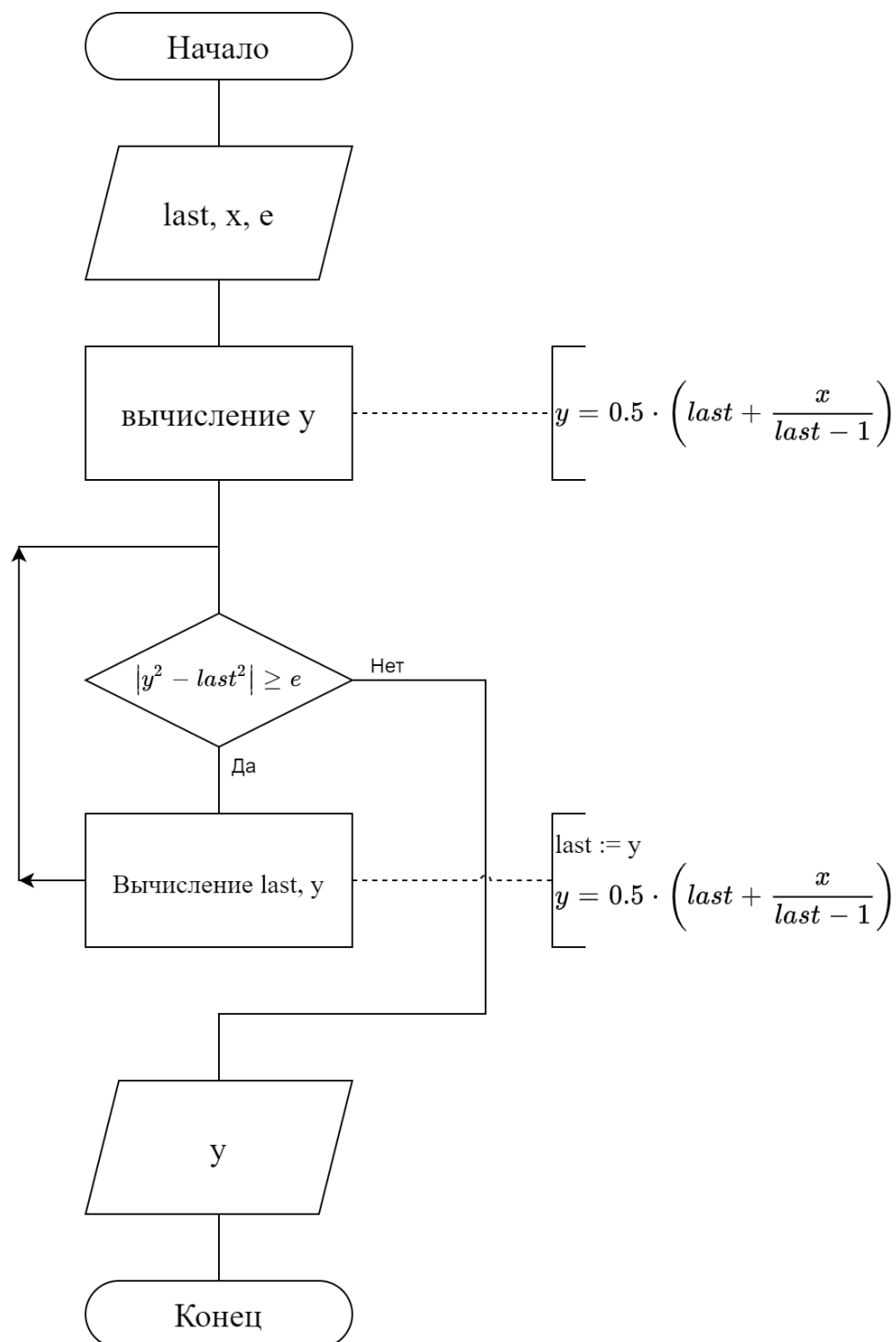


Рисунок 2 - Алгоритм решения задачи 2

Задача 3

Входные данные: $last$, x , e – действительные числа.

Выходные данные: y – действительное число.

Алгоритм решения задачи представлен на рисунке 3.

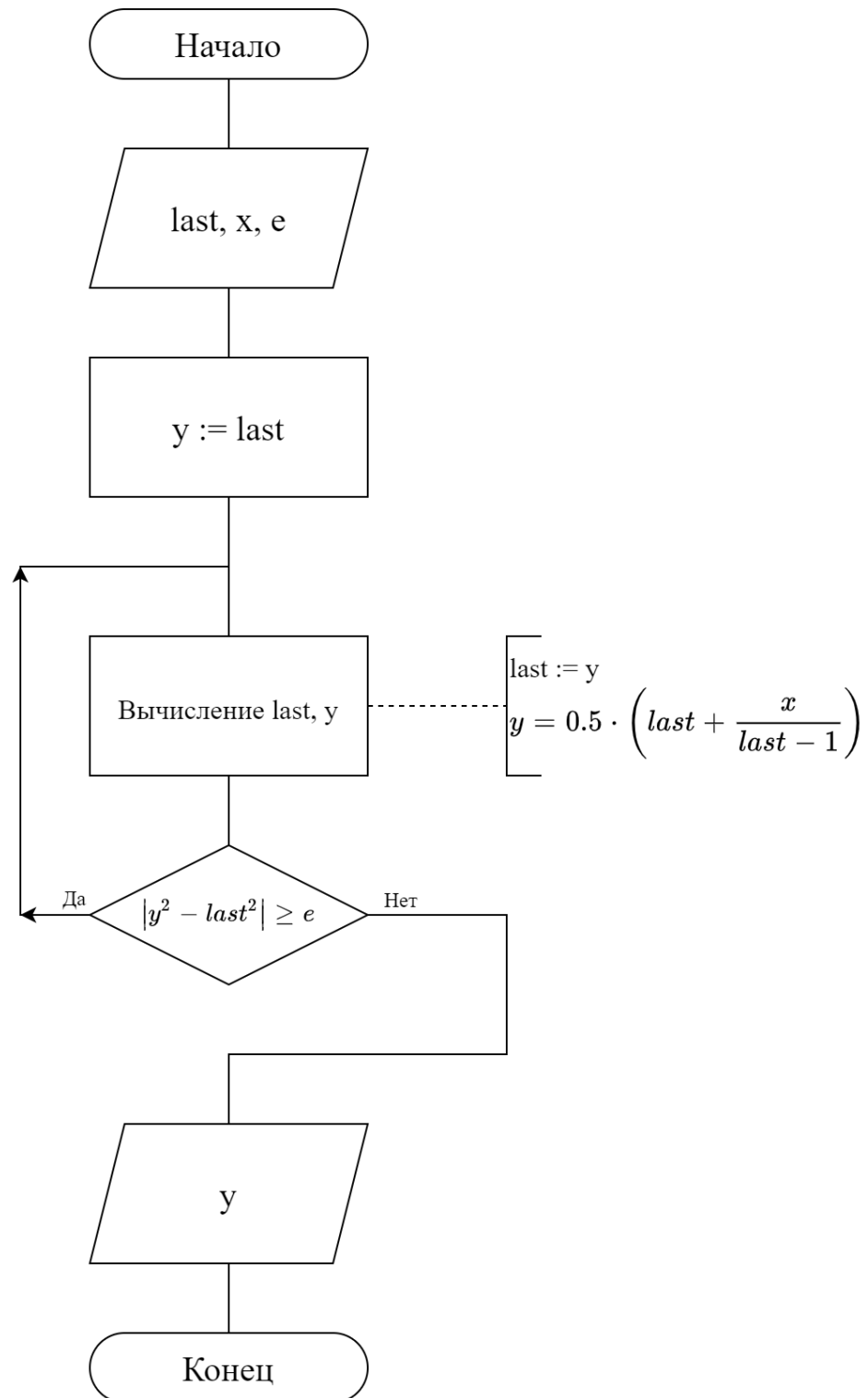


Рисунок 3 - Алгоритм решения задачи 3

Задача 4

Входные данные: k – целое число.

Выходные данные: n, m – пары целых чисел.

Алгоритм решения задачи представлен на рисунке 4.

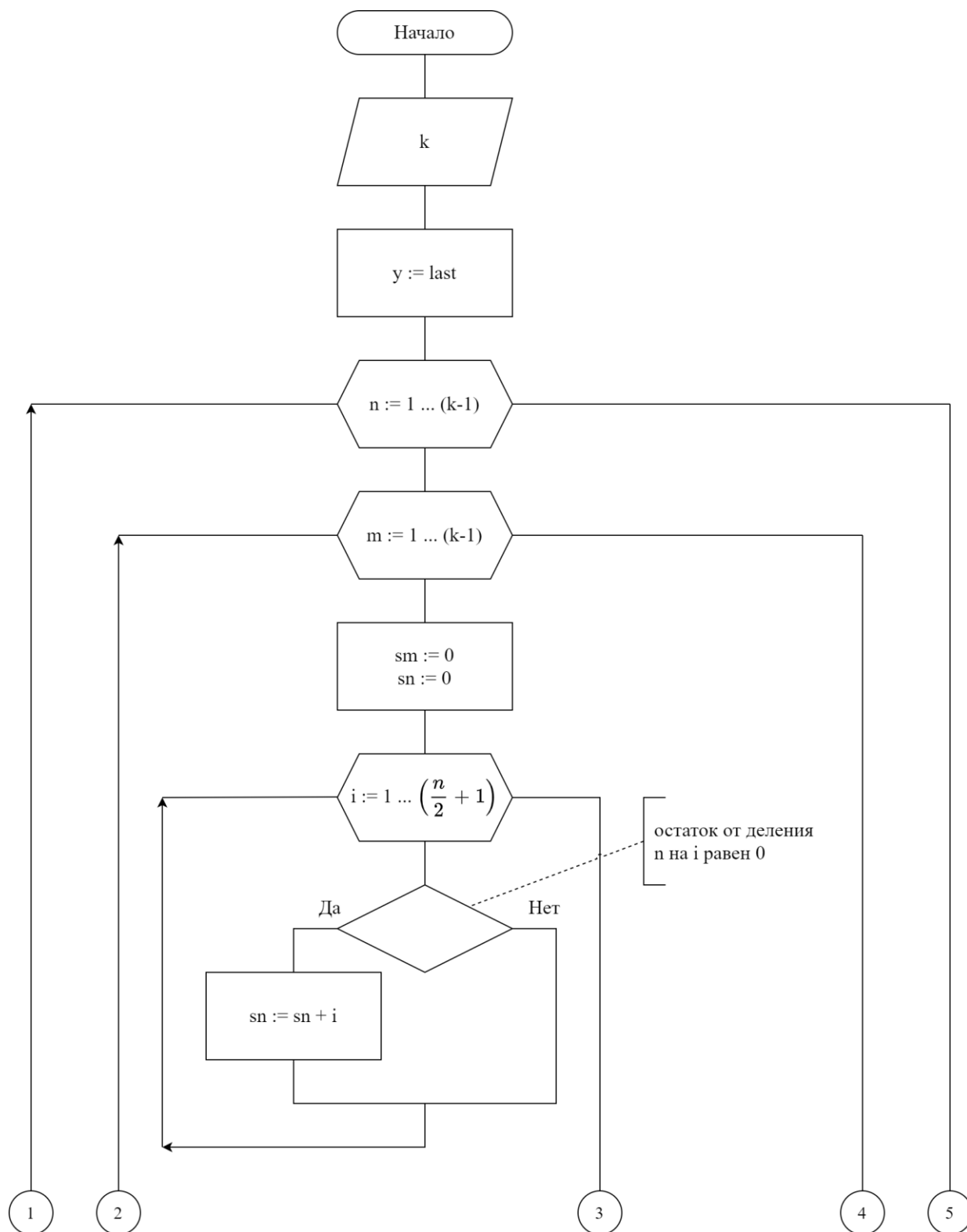


Рисунок 4 - Алгоритм решения задачи 4

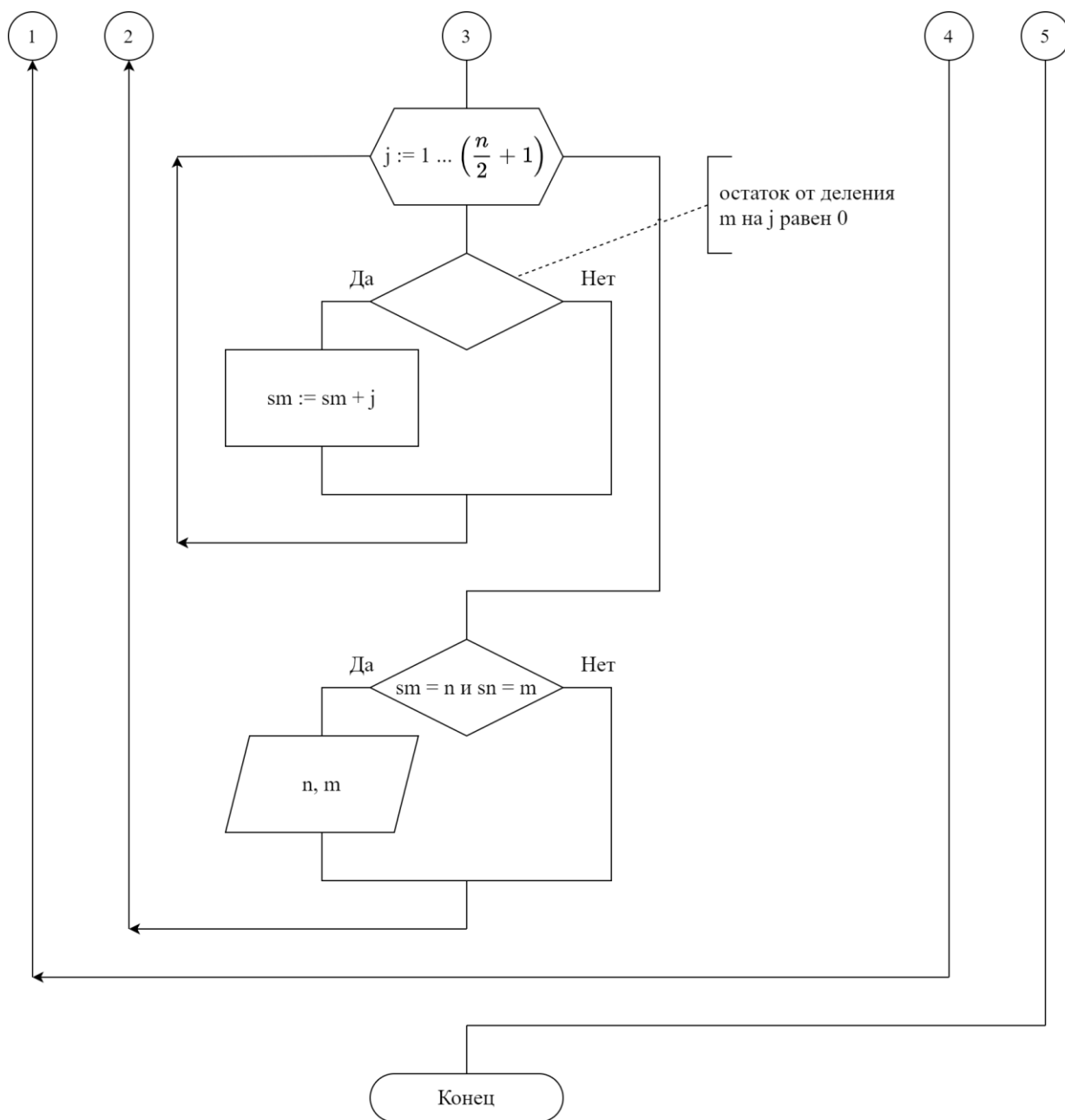


Рисунок 4 - Продолжение

Текст программы

Текст программы для решения задачи 1

```

#include <iostream>
using namespace std;

int main () {
    double n, sm = 0;
    cin >> n;

```



```

for (int i = 1; i <= n; i++) {
    sm += 1.0/i;
}

```

```

cout << sm;
}

```

Текст программы для решения задачи 2

```

#include <iostream>
#include <cmath>

```

```

using namespace std;

```

```

int main()
{
    double last, x, e;
    cin >> last, x, e;
    double y = 0.5 * (last + x / (last - 1));
    while ( abs(pow(y, 2) - pow(last, 2)) >= e ) {
        last = y;
        y = 0.5 * (last + x / (last - 1));
    }

    cout << y;
}

```

Текст программы для решения задачи 3

```

#include <iostream>
#include <cmath>

```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    double last, x, e;
```

```
    cin >> last, x, e;
```

```
    double y = last;
```

```
    do {
```

```
        last = y;
```

```
        y = 0.5 * (last + x / (last - 1));
```

```
    } while (abs(pow(y, 2) - pow(last, 2)) >= e);
```

```
    cout << y;
```

```
}
```

Текст программы для решения задачи 4

```
#include <iostream>
```

```
#include <cmath>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    int k;
```

```
    cin >> k;
```

```
    for (int n = 1; n < k; n++) {
```

```

for (int m = 1; m < k; m++) {
    int sn = 0;
    int sm = 0;

    for (int i = 1; i < n/2+1; i++) {
        if (n % i == 0) {
            sn += i;
        }
    }

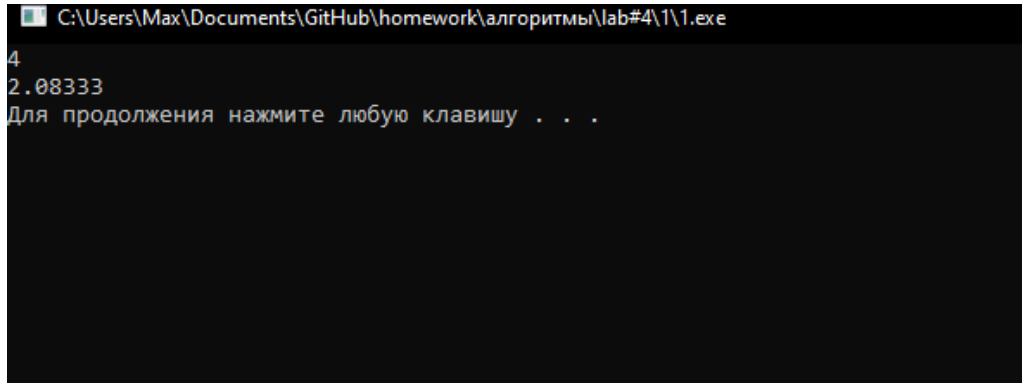
    for (int j = 1; j < m/2+1; j++) {
        if (m % j == 0) {
            sm += j;
        }
    }

    if (sm == n && sn == m) {
        cout << n << " " << m << endl;
    }
}
}
}

```

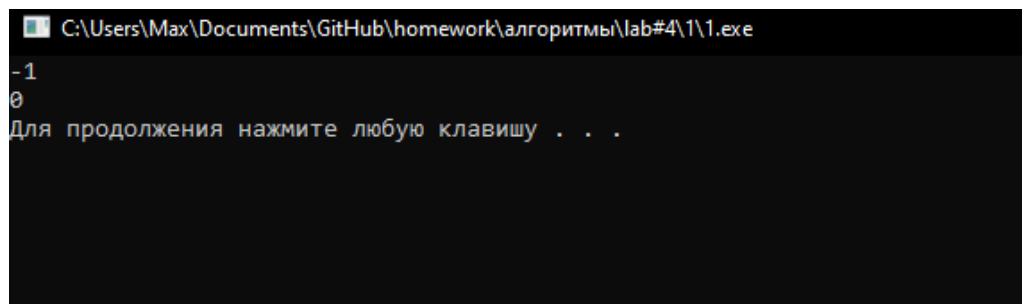
Тестирование программы

Тестирование задачи 1 представлено на рисунках 5, 6, 7



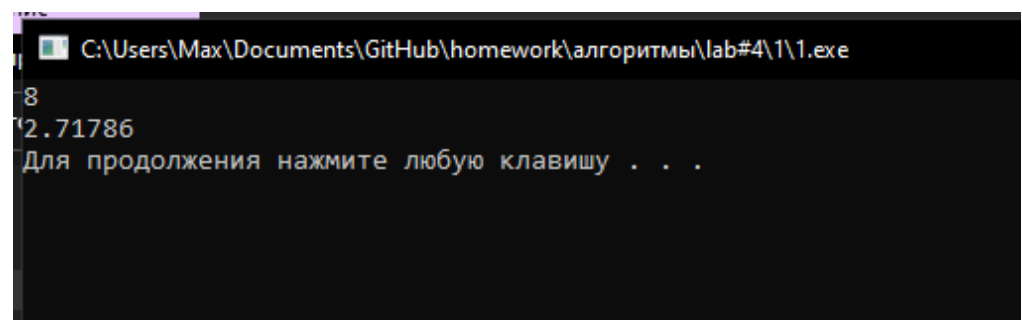
```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\1\1.exe
4
2.08333
Для продолжения нажмите любую клавишу . . .
```

Рисунок 5 - Тест 1 задачи 1



```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\1\1.exe
-1
0
Для продолжения нажмите любую клавишу . . .
```

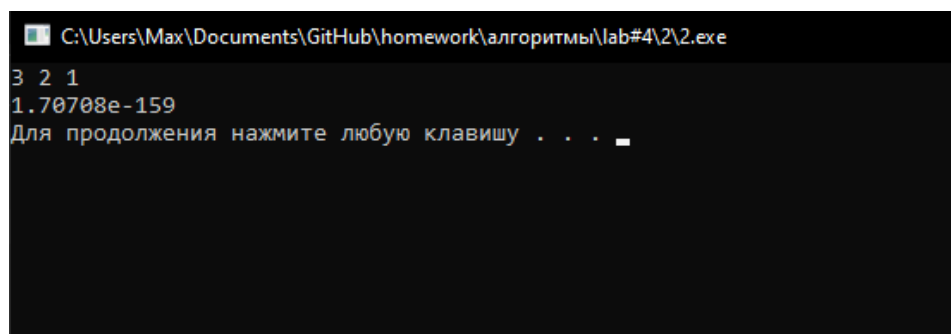
Рисунок 6 - Тест 2 задачи 1



```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\1\1.exe
8
2.71786
Для продолжения нажмите любую клавишу . . .
```

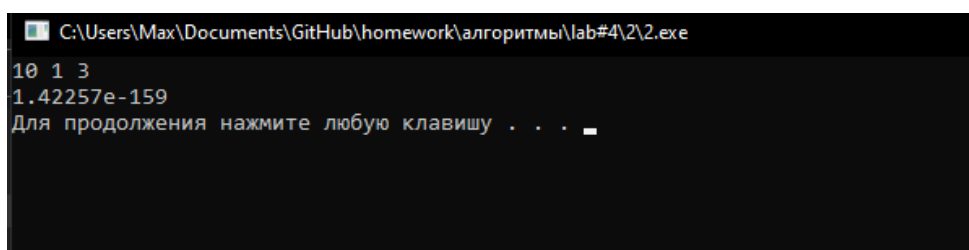
Рисунок 7 - Тест 3 задачи 1

Тестирование задачи 2 представлено на рисунках 8, 9, 10



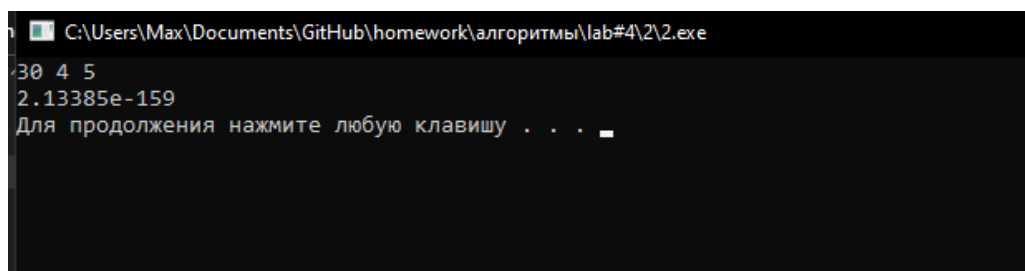
```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\2\2.exe
3 2 1
1.70708e-159
Для продолжения нажмите любую клавишу . . .
```

Рисунок 8 - Тест 1 задачи 2



```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\2\2.exe
10 1 3
1.42257e-159
Для продолжения нажмите любую клавишу . . .
```

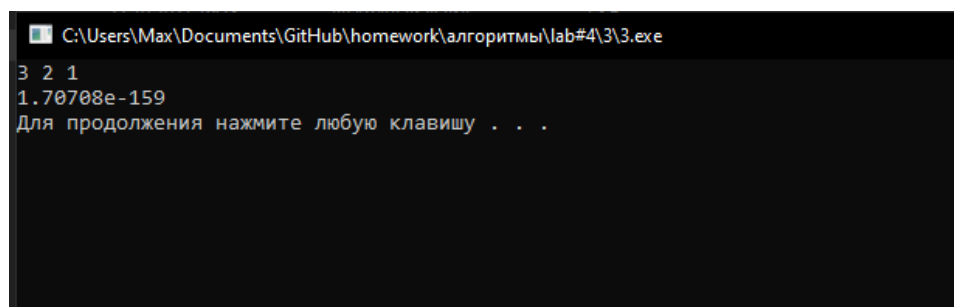
Рисунок 9 - Тест 2 задачи 2



```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\2\2.exe
30 4 5
2.13385e-159
Для продолжения нажмите любую клавишу . . .
```

Рисунок 10 - Тест 3 задачи 2

Тестирование задачи 3 представлено на рисунках 11, 12, 13



```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\3\3.exe
3 2 1
1.70708e-159
Для продолжения нажмите любую клавишу . . .
```

Рисунок 11 - Тест 1 задачи 3

```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\3\3.exe
10 1 3
1.42257e-159
Для продолжения нажмите любую клавишу . . .
```

Рисунок 12 - Тест 2 задачи 3

```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\3\3.exe
30 4 5
2.13385e-159
Для продолжения нажмите любую клавишу . . .
```

Рисунок 13 - Тест 3 задачи 3

Тестирование задачи 4 представлено на рисунках 14, 15, 16

```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\4\4.exe
50
6 6
28 28
Для продолжения нажмите любую клавишу . . .
```

Рисунок 14 - Тест 1 задачи 4

```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\4\4.exe
1000
6 6
28 28
220 284
284 220
496 496
Для продолжения нажмите любую клавишу . . .
```

Рисунок 15 - Тест 2 задачи 3

```
C:\Users\Max\Documents\GitHub\homework\алгоритмы\lab#4\4.exe
3000
6 6
28 28
220 284
284 220
496 496
1184 1210
1210 1184
2620 2924
2924 2620
Для продолжения нажмите любую клавишу . . .
```

Рисунок 16 - Тест 3 задачи 3